Engineering
SERIAL BIT STREAM ANALYSIS USING QUANTUM-DOT CELLULAR AUTOMATA

Jason Janulis	Douglas Tougaw*	Steve Henderson	Eric W. Johnson*
Valparaiso University	Valparaiso University	Valparaiso University	Valparaiso University
138 Gellersen Center	187 Gellersen Center	138 Gellersen Center	137 Gellersen Center
Valparaiso, IN 46383	Valparaiso, IN 46383	Valparaiso, IN 46383	Valparaiso, IN 46383
Phone: (219) 464-5054	Phone: (219) 464-5027	Phone: (219) 464-5054	Phone: (219) 464-5176
Fax: (219) 464-5065	Fax: (219) 464-5065	Fax: (219) 464-5065	Fax: (219) 464-5065
Jason.Janulis@valpo.edu	Doug.Tougaw@valpo.edu	Steve.Henderson@valpo.edu	Eric.Johnson@valpo.edu

Quasi-adiabatically switched quantum-dot cellular automata (QCA) devices present the opportunity to extend our efforts from the implementation of combinational logic devices to more useful sequential logic devices. One very important application of sequential logic is in the recognition of patterns in serial bit streams. This is important, for example, in Internet applications, where particular bit patterns are designated as “sentinel” characters that indicate a particular action should be taken.

The foundation of a serial bit stream analyzer is a shift register, which can be implemented very easily using quasi-adiabatically switched QCA devices. In addition to the shift register, the device will require a multiple-bit comparator, which has not yet been demonstrated in a QCA architecture.

We will present a multiple-bit serial stream analyzer that combines the functions of the shift register and the comparator. This device will be analyzed first using behavioral and structural models developed specifically for this project, then its correct quasi-adiabatic behavior will be demonstrated using well established quantum mechanical models.

We will show that the proposed device can be used to detect any arbitrary bit pattern appearing in a serial stream of data applied to its serial input. The bit pattern to be detected can be changed using device inputs, and a successful match will be indicated by asserting an output.